



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

09/770,705

01/26/2001

Christopher S. Gouge

MS155721.2

6801

27195

7590

01/10/2005

AMIN & TUROCY, LLP  
24TH FLOOR, NATIONAL CITY CENTER  
1900 EAST NINTH STREET  
CLEVELAND, OH 44114

EXAMINER

INGBERG, TODD D

ART UNIT

PAPER NUMBER

2124

DATE MAILED: 01/10/2005

Please find below and/or attached an Office communication concerning this application or proceeding.

<b>Office Action Summary</b>	<b>Application No.</b> 09/770,705	<b>Applicant(s)</b> GOUGE ET AL.	
	<b>Examiner</b> Todd Ingberg	<b>Art Unit</b> 2124	

**-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 27 September 2004.
- 2a) ☐ This action is **FINAL**.                      2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-24 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-24 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 1/26/2001 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All    b) ☐ Some \*    c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).
- \* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892)  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### DETAILED ACTION

Claims 1 – 24 have been examined.

1. A Request for Continued Examination (RCE) was established on September 27, 2004.

#### *Claim Rejections - 35 USC § 101*

2. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

Claims 1- 24 are rejected under 35 U.S.C. 101 because the claimed invention lacks patentable utility.

#### Claim 1

A system that facilitates software installation **executing on a computer and stored on a computer readable medium** comprising:

a transformation component that receives one or more configurable data elements, and one or more non-configurable data elements describing the one or more configurable data elements; and

a merge component that employs one or more transformation instructions that configures the one or more configurable data elements to facilitate the installation of the one or more configurable data elements into at least one target data set.

#### Claim 9

A data interpretation system **executing on a computer and stored on a computer readable medium**, comprising: a data interpretation component that

receives one or more configurable data elements from a configurable module wherein the one or more data elements include configuration information related to installing the one or more data elements into a software program; applies one or more transformation instructions from the configurable module to the configurable data elements to configure the configurable data elements, and installs the configurable data elements into a target data set based at least in part upon the configuration information,

#### Claim 13

A method for installing **executing on a computer and stored on a computer readable medium** a configurable data set into a target data set, comprising:

obtaining one or more data elements from a configurable module; including metadata that describes the configuration options of the configurable data set obtaining one or more transformation instructions from the configurable module; and applying the one or more

Art Unit: 2124

transformation instructions to a copy of the one or more data elements from the configurable module to configure the one or more data elements for installation into the target data set.

Claim 17

A method for creating a configurable data module **executing on a computer and stored on a computer readable medium**, comprising: creating a configurable data set having one or more configurable data elements; and creating one or more data structures containing information associated with one or more configurable data elements, and displaying the information to a software program to facilitate installing the configurable data set into the software program.

Claim 22

A system for installing **executing on a computer and stored on a computer readable medium**, a configurable data set into a target data set, comprising: a configurable module having configurable data elements representing a configurable data set and non-configurable data elements representing a portion of the configurable data set; a user interface for selecting which configurable data element to modify; and a data interpretation system for receiving the configurable and non-configurable data elements from the configurable module and applying the transformation instructions applicable to the user selections associated with the configurable data elements to enable installing the configurable data set into the target data set.

A data packet is a data structure.

Claim 23

A data packet adapted to be transmitted between at least two computer processes, comprising: a configurable module having: one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; one or more non-configurable data elements describing the one or more configurable data elements; and one or more transformation instructions that facilitate configuring the one or more configurable data elements, wherein the instructions are employed to facilitate installation of the one or more configurable data elements into a target data set.

Claim 24 is drawn to a data structure.

Claim 24

A computer readable medium having stored thereon a data structure, comprising: a first data field containing one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; a second data field containing one or more non-configurable data elements describing the one or more configurable data elements; and a third data field containing one or more transformation instructions that facilitate configuring the one or more configurable data elements, to load the configurable data elements into a software program.

***Claim Rejections - 35 USC § 102***

3. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

4. Claims 1 – 24 are rejected under 35 U.S.C. 102(b) as being anticipated by Microsoft's Visual C++ version 5.0 as documented in the text book, "Beginning Visual C++ 5", by Ivor Horton, published March 19, 1997. Referred to as **DLL**.

***Term's in the Art***

The following are terms in art the one of ordinary skill in the art should have knowledge of at the time of invention and the interpretations given during prosecution.

**A. Dynamic-Link Library (DLL)** - [Microsoft Computer Dictionary, page 166, published September 19, 1997]

A feature of the Microsoft Windows family of operating systems and OS/2 that allows executable routines to be stored separately as files with the DLL extensions and to be loaded only when needed by a program. A dynamic-link library has several advantages. First, it does not consume any memory until it is used. Second, because a dynamic-link library is a separate file, a programmer can make corrections or improvements to only that module without affecting the operation of the calling program or any other dynamic-link library. Finally, a programmer can use the same dynamic-link library with other programs.

**B. Dynamic Link Library (DLL)** as defined by the IBM Dictionary page 225. "A file containing executable code and data bound to a program at load time, or run time, rather than during linking. The code and data in a dynamic link library can be shared by several applications simultaneously."

Art Unit: 2124

**Claim 1**

**DLL** anticipates a system that facilitates software installation comprising: a transformation component that receives one or more configurable data elements, and one or more non-configurable data elements describing the one or more configurable data elements; and a merge component that employs one or more transformation instructions that configures the one or more configurable data elements to facilitate the installation of the one or more configurable data elements into at least one target data set.

**Examiner's Response**

By definition as provided in the section Term's in the Art, a DLL adds data and/or executable code at link or runtime and as the name states the file is linked. The data elements are the different portions of the program which the data elements or functions may or may not be updated replaced by the linkages to the DLL. DLL on page 719 figure shows a library of DLLs and the program which only a portion is updated with the linkages to the DLL replacing the function. The functions in the program correlate to the functions in the DLL and the data elements relate to data in the DLL. the transformation is the runtime update of the linkages causing the installation of the software to transform the runtime environment. DLL teaches many uses of DLLs from pages 715 to 734. Pages 715 to 716 provide an overview of DLLs, page 717 to 719 teach How DLLs Work 720 to 722 teach the content of a DLL, pages 722 – 729 show the tool for performing the task. Pages 729 to 733 clearly ties into the definition by examples.

**Claim 2**

The system of claim 1, wherein the one or more configurable data elements are stored in a data structure associated with the configurable module.

**Examiner's Response**

The DLL is a file files inherently have a data structure in view of the rejection for claim 1.

**Claim 3**

The system of claim 2, wherein the data structure is a metadata item description table.

**Examiner's Response**

The metadata (data about data by definition) is the address of the linkages which control the running of the program. Most college text books refer to this as a program look up table. the presence is seen in the rejection for claim one with the linkages. The affect of the DLL on the metadata is the linkages instruct the merging of components to perform replacements as illustrated in the rejection for claim 1.

**Claim 4**

The system of claim 1, wherein the one or more non-configurable data elements are stored in a data structure associated with the configurable module.

**Examiner's Response**

A non configurable data element is a portion of the program not changed by the DLL as described in the rejection for claim 1.

**Claim 5**

Art Unit: 2124

The system of claim 4, wherein the one or more non-configurable data elements are stored in a metadata item description table.

**Examiner's Response**

The portions of the program not changed by the DLL inherently are stored in the program lookup table (metadata).

**Claim 6**

The system of claim 1, wherein the one or more transformation instructions are stored in a data structure associated with the configurable module.

**Examiner's Response**

The actual transformation instructions are the linkages which are addresses to the beginning and end of functions.

**Claim 7**

The system of claim 6, wherein the one or more transformation instructions are stored in a transformation instruction table.

**Examiner's Response**

The presence of a function to be replaced in the figure on 719 is evident of at least one transformation.

**Claim 8**

The system of claim 2 wherein the data structure includes at least one of a name of a configurable data element, and a semantic meaning for the configurable data element.

**Examiner's Response**

The name is the name of the function being replaced in the example in claim 1. The semantic meaning being the linking of executable code and/or data at link or runtime must be compatible in order to run. This is deemed inherent.

**Claim 9**

DLL anticipates a data interpretation system, comprising: a data interpretation component that receives one or more configurable data elements from a configurable module wherein the one or more data elements include configuration information related to installing the one or more data elements into a software program; applies one or more transformation instructions from the configurable module to the configurable data elements to configure the configurable data elements, and installs the configurable data elements into a target data set based at least in part upon the configuration information,

**Examiner's Response**

As per claim 1 the definition of a DLL is the ability to add data and/or executable code at link or runtime. The example in claim 1 is of a code replacement. the mechanism to change the address to a data element is the same and is inherent in the rejection of claim 1 by definition. Page 729 to 730, adding variables from a DLL is covered.

**Claim 10**

Art Unit: 2124

The system of claim 9, further comprising a user interface to enable a user to query the configurable module to determine which of the one or more data elements are configurable.

**Examiner's Response**

Page 725 shows the user interface to create DLLs and search.

**Claim 11**

The system of claim 9 further comprising: a merging component adapted to receive one or more updated configurable data elements from the data interpretation component and adapted to provide the one or more updated configurable data elements to a target data set.

**Examiner's Response**

As per the rejection for claim 1.

**Claim 12**

The system of claim 9 further comprising: an authoring schema that describes a configurable module.

**Examiner's Response**

DLL Chapter 18 is dedicated to being able to write your own DLLs. Page 717 to 719 explain How DLLs Work.

**Claim 13**

DLL anticipates a method for installing a configurable data set into a target data set, comprising: obtaining one or more data elements from a configurable module; including metadata that describes the configuration options of the configurable data set obtaining one or more transformation instructions from the configurable module; and applying the one or more transformation instructions to a copy of the one or more data elements from the configurable module to configure the one or more data elements for installation into the target data set.

**Examiner's Response**

The rejection for claim 1 covers the limitations for claim 13.

**Claim 14**

The method of claim 13, further comprising: identifying a target data set; and inserting the updated data elements into the target data set.

**Examiner's Response**

DLL pages 729 to 730 show adding a variable.

**Claim 15**

The method of claim 13 further comprising: presenting one or more configuration options to a user; accepting one or more configuration selections from the user; and selectively configuring the one or more data elements based on the user's configuration selections.

**Examiner's Response**

Page 728 shows several configuration options in the figure link incrementally, ignore all default libraries etc.

**Claim 16**



Art Unit: 2124

A computer readable medium containing computer executable instructions operable to perform the method of claim 13.

**Examiner's Response**

DL by definition.

**Claim 17**

DLL anticipates a method for creating a configurable data module, comprising: creating a configurable data set having one or more configurable data elements; and creating one or more data structures containing information associated with one or more configurable data elements, and displaying the information to a software program to facilitate installing the configurable data set into the software program.

**Examiner's Response**

The rejection for claim explains the details of DLLs and page 725 shows a user interface. Also see page 727 the Sketcher program for determining the addresses for linkage.

**Claim 18**

The method of claim 17 wherein creating a configurable data set includes: identifying one or more attributes of the one or more data elements; and establishing one or more default values for the attributes of the one or more data elements.

**Examiner's Response**

DLL, page 724 top of the page ExtDLLEExample.cpp (key is the .cpp) contains the information.

**Claim 19**

The method of claim 17 wherein creating the one or more data structures further comprises: identifying one or more locations within a data set that are configurable; identifying one or more configuration options; creating one or more instructions concerning how to configure the one or more locations; and storing the instructions in the one or more data structures.

**Examiner's Response**

DLL, page 724 top of the page ExtDLLEExample.def (key is the .def) contains the information.

**Claim 20**

The method of claim 19, wherein the one or more data structures are stored in the configurable data module.

**Examiner's Response**

DLL, page 724 the file names at the top of the page.

**Claim 21**

A computer readable medium containing computer executable instructions operable to perform the method of claim 17.

**Examiner's Response**

DLL by definition.

**Claim 22**

Art Unit: 2124

DLL anticipates a system for installing a configurable data set into a target data set, comprising: a configurable module having configurable data elements representing a configurable data set and non-configurable data elements representing a portion of the configurable data set; a user interface for selecting which configurable data element to modify; and a data interpretation system for receiving the configurable and non-configurable data elements from the configurable module and applying the transformation instructions applicable to the user selections associated with the configurable data elements to enable installing the configurable data set into the target data set.

**Examiner's Response**

See the rejection for claim 1.

**Claim 23**

DLL anticipates a data packet adapted to be transmitted between at least two computer processes, comprising: a configurable module having: one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; one or more non-configurable data elements describing the one or more configurable data elements; and one or more transformation instructions that facilitate configuring the one or more configurable data elements, wherein the instructions are employed to facilitate installation of the one or more configurable data elements into a target data set.

**Examiner's Response**

See the rejection for claim 1.

**Claim 24**

DLL anticipates a computer readable medium having stored thereon a data structure, comprising: a first data field containing one or more configurable data elements, wherein one or more default values for the one or more configurable data elements are available; a second data field containing one or more non-configurable data elements describing the one or more configurable data elements; and a third data field containing one or more transformation instructions that facilitate configuring the one or more configurable data elements, to load the configurable data elements into a software program.

**Examiner's Response**

The rejection for claim 1 teaches the details of DLLs. Default values can be interpreted the linkages or possibly the importing of Symbols page 730 to 733.

***Conclusion***

5. Microsoft Visual Source Safe (VSS) version 5.0 as documented in the text book "Web Management with Microsoft Visual SourceSafe 5.0", by Steven Banick et al., published by QUE, 1997. Pages 112 – 116 cover merging components in a configuration management environment.

***Correspondence Information***

Art Unit: 2124

Any inquiry concerning this communication or earlier communications from the examiner should be directed to **Todd Ingberg** whose telephone number is **(571) 272-3723** (as of October 23, 2004).

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, **Kakali Chaki** can be reached on **(571) 272-3719**. Please, note that as of August 4, 2003 the **Official FAX number** changed to **(703) 872-9306**.

Also, be advised the United States Patent Office **new address** is

Post Office Box 1450

Alexandria, Virginia 22313-1450

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the receptionist whose telephone number is **(703) 305-3900**.

A handwritten signature in black ink, appearing to read 'Todd Ingberg', with a long horizontal flourish extending to the right.

**Todd Ingberg**  
Primary Examiner  
Art Unit 2124  
December 27, 2004